

Características	Programacion Extrema	Scrum	Kanban	Scrumban
Objetivo principal	Mejora continua del código y calidad mediante prácticas técnicas rigurosas.	Optimizar la gestión de trabajo en ciclos cortos, con un enfoque en la comunicación y entrega del producto.	Mejorar la eficiencia del flujo de trabajo y la entrega continua sin iteraciones fijas.	Combinar la planificación estructurada de Scrum con la flexibilidad y flujo continuo de Kanban.
Enfoque	Calidad del código y capacidad de adaptación rápida.	Gestión de proyectos y colaboración del equipo.	Gestión del flujo de trabajo en tiempo real.	Equilibrar estructura (Scrum) con flexibilidad en el flujo de trabajo (Kanban).
duración de iteraciones	Cortas, típicamente de 1-2 semanas.	Sprints de 2-4 semanas.	Sin iteraciones fijas, flujo continuo.	Iteraciones flexibles, los sprints se usan solo si es necesario.
Roles Clave	Cliente, Desarrollador, Coach.	Product Owner, Scrum Master, Equipo de Desarrollo.	No hay roles definidos formalmente.	Scrum Master y Product Owner pueden ser opcionales; roles más flexibles.
Planificación	Planificación diaria y ajustada a los cambios según sea necesario.	Planificación al inicio de cada sprint, con backlog.	Flujo continuo, basado en la demanda.	Planificación de trabajo en base al backlog, pero con posibilidad de ajuste continuo.
Entrega del producto	Entregas frecuentes, normalmente después de cada iteración.	Entregas al final de cada sprint.	Entrega continua, sin necesidad de esperar iteraciones.	Entrega continua, con revisiones más estructuradas si es necesario.
Priorización del trabajo	Según necesidades del cliente y requisitos urgentes.	Se prioriza el backlog con base en el valor para el negocio.	Las tareas se gestionan según el límite de trabajo en progreso (WIP).	Se prioriza el backlog, pero las tareas se gestionan con flexibilidad según los límites de WIP.

control del progreso	Revisión constante a través de pruebas y refactorización continua.	Revisiones diarias (Daily Scrum) y reuniones de revisión al final del sprint.	Visualización del trabajo mediante un tablero Kanban que muestra el progreso.	Se utilizan tableros Kanban y ceremonias opcionales (Daily Stand-ups, Retrospectivas).
adaptabilidad al cambio	Muy adaptable; se promueve la respuesta rápida a los cambios mediante iteraciones cortas.	Se adapta al cambio entre sprints, permitiendo reordenar el backlog.	Altamente flexible, se adapta en tiempo real a nuevas demandas.	Altamente flexible, combina la estructura de Scrum con la adaptabilidad de Kanban.
medición del progreso	Pruebas automáticas, métricas de calidad y velocidad de desarrollo.	Velocidad del equipo, burndown charts y retrospectión.	Tiempo de ciclo, límites de WIP, y tiempo total para completar tareas.	Velocidad del equipo, métricas de flujo (WIP, tiempo de ciclo).
metodología de trabajo	Programación en parejas, integración continua, refactorización, pruebas unitarias frecuentes.	Daily Stand-up, Sprint Planning, Sprint Review, Sprint Retrospective.	Tablero Kanban, límite de WIP, mejora continua.	Uso de tableros Kanban con elementos de Scrum como sprints opcionales y retrospectión.
fortalezas	Alta calidad técnica del código, excelente adaptabilidad, comunicación continua con el cliente.	Claridad en los roles, mejora continua, planificación a corto plazo, enfoque en objetivos claros por sprint.	Máxima flexibilidad, visualización del progreso, fluidez en la entrega de trabajo.	Combina la estructura de Scrum con la flexibilidad de Kanban, adaptable a proyectos y equipos dinámicos.
desafíos	Puede ser difícil de implementar en equipos grandes o en entornos donde los cambios son limitados.	Puede ser rígido si los sprints no se gestionan adecuadamente, requiere dedicación a las ceremonias.	Puede volverse caótico si no se gestionan bien los límites de WIP, falta de estructura en equipos sin experiencia.	Puede generar confusión si no se gestiona bien la mezcla de prácticas; equipos necesitan experiencia en ambas metodologías.

cuando usar	Proyectos que requieren cambios frecuentes, alta calidad del código y equipos muy colaborativos.	Proyectos donde se requiere una estructura organizada con iteraciones y objetivos claros.	Proyectos con flujo continuo de trabajo y alta variabilidad en las demandas.	Ideal para equipos que buscan flexibilidad pero también desean mantener un poco de estructura formal.
--------------------	--	---	--	---